



KOREAN PATENT ABSTRACTS

(11)Publication number: 1020010017953 A
 (43)Date of publication of application: 05.03.2001

(21)Application number: 1019990033711
 (22)Date of filing: 16.08.1999

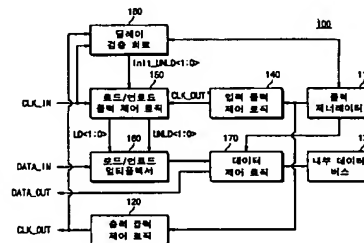
(71)Applicant: SAMSUNG ELECTRONICS CO., LTD.
 (72)Inventor: SHIN, YEONG MIN

(51)Int. Cl. G06F 1/12

(54) CLOCK FORWARDING CIRCUIT HAVING CHARACTERISTICS FOR DETECTING AUTOMATIC CLOCK DELAY AND SETTING INITIAL PARAMETER

(57) Abstract:

PURPOSE: A clock forwarding circuit having characteristics for detecting an automatic clock delay and setting an initial parameter is provided to transmit data between circuits without error and the affection due to the delay in interface clocks, by automatically detecting the delay, and by forwarding the clocks corresponding to the detected delay.



CONSTITUTION: A clock generator(110) produces the bit clock for inputting and outputting data. An output clock control logic(120) controls the clock signal from the clock generator(110) to generate and apply an output clock signal(CLK_OUT) to a slave circuit. The data(DATA_OUT), transmitted from the internal data bus(130) of a master circuit, is output to the slave circuit via a data control logic(170). The slave circuit receives the clock signal(CLK_OUT) and the data(DATA_OUT), and outputs an input clock signal(CLK_IN) and an input data(DATA_IN) to a clock forwarding circuit(100) in the master circuit. A delay detection circuit(180) automatically detects the delay and applies the initial parameter(init_UNLD) corresponding to the delay to a load/unload clock control logic (150).

COPYRIGHT 2001 KIPO

Legal Status

Date of final disposal of an application (20020131)

Patent registration number (1003360410000)

Date of registration (20020426)

공개특허 제2001-17953호(2001.03.05) 1부.

[첨부그림 1]

특2001-0017953

(19) 대한민국특허청(KR) (12) 공개특허공보(A)

(51) Int. Cl.⁸
G06F 1/12

(11) 공개번호 특2001-0017953
(43) 공개일자 2001년03월05일

(21) 출원번호	10-1999-0033711
(22) 출원일자	1999년08월16일
(71) 출원인	삼성전자 주식회사 윤종용 경기 수원시 팔달구 매탄3동 416
(72) 발명자	신영민 서울특별시마포구성산1동251-23번지
(74) 대리인	임창현

심사청구 : 있음

(54) 자동 클럭 딜레이 검출 및 초기 파라미터 셋팅 특성을 가진 클럭 포워딩 회로

요약

최근 들어, CPU(central processing unit)와 같은 마스터 회로(master circuit)의 클럭 속도가 증가함에 따라 반도체 메모리 장치 및 시스템 버스와 같은 슬레이브 회로의 클럭 속도가 증가하고 있다. 슬레이브 회로는 클럭 및 데이터 경로들을 통해 마스터 회로와 외부 인터페이스를 수행하는데, 상기 회로들 사이에 인터페이스 되는 클럭 신호들에는 마더보드(mother board)의 구조에 의한 딜레이(delay)가 존재한다. 회로들의 클럭 속도들이 증가함에 따라 데이터를 로드/언로드(load/unloading)하기 위한 오퍼레이팅 마진(operating margin)이 줄어든다. 따라서, 상기 회로들 사이에 결합 없이 데이터를 전송하기가 어렵다. 따라서, 회로들 사이의 데이터 전송을 안정화하기 위해, 본 발명에 의한 마스터 회로는 상기 딜레이를 자동적으로 검출하고, 상기 딜레이에 상응하는 클럭들을 포워딩하기(앞서 전송하기) 위한 클럭 포워딩 회로(clock forwarding circuit)를 포함한다. 상기 회로들 사이의 데이터 전송은, 클럭들의 포워딩에 의해서 딜레이에 영향을 받지 않고 결합 없이 수행될 수 있다.

도면

도 4

도면의 주요부분에 대한 부호의 설명

도 1은 마스터 회로와 슬레이브 회로를 구비한 디지털 시스템의 구성을 개략적으로 보여주기 위한 블록도;
도 2A-2C는 도 1에 도시된 마스터 회로의 입력 및 출력 클럭 신호들의 관계를 보여주기 위한 타이밍도;
도 3은 도 1에 도시된 마스터 회로의 클럭 포워딩 방법에 의한 입력 및 출력 클럭 신호들의 관계를 보여주기 위한 타이밍도;
도 4는 본 발명에 의한 마스터 회로의 클럭 포워딩 회로를 개략적으로 보여주기 위한 블록도;
도 5는 도 4에 도시된 딜레이 검출 회로를 보여주기 위한 회로도;
도 6은 도 5에 도시된 딜레이 검출 회로의 동작을 보여주기 위한 타이밍도;
도 7은 도 5에 도시된 딜레이 검출 회로에 의해서 검출된 최대 딜레이에 대응되는 초기 파라미터를 보여주기 위한 도면;
도 8은 도 4에 도시된 로드/언로드 클럭 제어 로직을 보여주기 위한 회로도;
도 9는 도 8에 도시된 로드/언로드 클럭 제어 로직의 동작을 보여주기 위한 타이밍도;
도 10은 도 4에 도시된 로드/언로드 멀티플렉서를 보여주기 위한 회로도; 그리고
도 11은 도 4에 도시된 클럭 포워딩 회로의 동작을 보여주기 위한 타이밍도.
도면의 주요 부분에 대한 부호의 설명
100 : 클럭 포워딩 회로 110 : 클럭 제너레이터
120 : 출력 클럭 제어 로직 130 : 내부 데이터 버스
140 : 입력 클럭 제어 로직 150 : 로드/언로드 클럭 제어 로직

160 : 로드/언로드 멀티플렉서 170 : 데이터 제어 로직
180 : 딜레이 검출 회로

발명의 상세한 설명

발명의 목적

발명이 속하는 기술분야 및 그 분야의 종래기술

본 발명은 디지털 정보처리용 집적회로(digital information processing IC)에 관한 것으로, 좀 더 구체적으로는 자동 클럭 딜레이 검출 및 초기 파라미터 셋팅 특성을 가진 클럭 포워딩 회로에 관한 것이다.

도 1은 마이크로프로세서(예를 들면 CPU)와 같은 마스터 회로(master circuit; 10), 그리고 반도체 메모리 장치 및 시그널 버스(또는 시스템 버스)와 같은 슬레이브 회로(slave circuit; 20)를 포함하는 시스템을 개략적으로 보여주기 위한 블록도이다.

도 1을 참조하면, 슬레이브 회로(20)는 클럭 및 데이터 경로를 통해서 마스터 회로(10)와 외부 인터페이스를 수행한다. 마스터 회로(10)는 클럭 신호(CLK_OUT)와 데이터(DATA_OUT)를 슬레이브 회로(20)로 발생한다. 슬레이브 회로(20)는 상기 클럭 신호(CLK_OUT)와 데이터(DATA_OUT)를 받아들이고, 클럭 신호(CLK_IN)와 데이터(DATA_IN)를 마스터 회로(10)로 발생한다. 여기서, 입력 클럭 신호(CLK_IN)는 출력 클럭 신호(CLK_OUT)의 피드백 클럭 신호(feedback clock signal)이다. 마스터 회로(10)는, 슬레이브 회로(20)로부터의 데이터(DATA_IN)를 로딩하는데 입력 클럭 신호(CLK_IN)를 사용하고, 로딩된 입력 데이터(DATA_IN)를 내부에서 처리하는데 출력 클럭 신호(CLK_OUT)를 사용한다.

도 2A-2C는 도 1에 도시된 마스터 회로(10)의 입력 및 출력 신호들의 관계를 보여주는 타이밍도이다. 도면을 참조하면, 클럭 신호(CLK_IN, CLK_OUT) 사이에는 딜레이(delay)가 존재하는데, 이는 마스터 회로(10)와 슬레이브 회로(20)를 포함하는 마더보드(motherboard)의 구성에 기인한다. 이와 같은 딜레이는, 회로들의 낮은 클럭 속도 때문에 과거에는 아무런 문제도 발생시키지 않았다. 도 2A에 도시된 바와 같이, 낮은 클럭 속도는 마스터 회로(10)의 데이터 로딩/loading 및 언로딩/unloading 동작들을 위해 충분한 오버레이팅 마진(operating margin)을 보장한다. 그러나, 최근 들어 마스터 회로(10) 및 슬레이브 회로(20)들의 클럭 속도가 높아지는 경향이 있다. 도 2B에 도시된 바와 같이, 마스터 회로(10) 및 슬레이브 회로(20)의 클럭 속도가 증가함에 따라 오버레이팅 마진은 감소한다. 그 결과, 슬레이브 회로(20)에서 마스터 회로(10)로의 데이터 전송에 오류가 발생할 수 있다. 도 2C를 참조하면, 만약 오버레이팅 마진을 보장할 수 없을 정도로 클럭 속도가 빨라지면, 상기 데이터(DATA_IN)를 마스터 회로(10)로 안전하게 전송하는 것은 더욱 어렵게 된다. 오버레이팅 마진 없이 출력 클럭 신호(CLK_OUT)가 입력 클럭 신호(CLK_IN)보다 앞서기 때문에, 마스터 회로(10)의 데이터 언로딩 동작은 데이터 로딩 동작 이후에 수행될 수 없게 되는 것이다. 특히, 고성능 컴퓨터 시스템에 있어서, 이러한 인터페이스 문제를 피하는 것은 처리 속도(processing speed)가 증가함에 따라 어려워지고 있다. 이러한 문제를 해결하기 위한 한 방법이 바로 클럭 포워딩 방법(clock forwarding method)이다.

도 3은 도 1에 도시된 마스터 회로의 클럭 포워딩 방법에 의한 입력 및 출력 클럭 신호들의 관계를 보여주는 타이밍도이다. 이 방법에서, 딜레이에 상응하는 몇몇 클럭 주기들은 도 1에 도시된 클럭 포워딩 회로(100)에 의해 출력 클럭 신호(CLK_OUT)에 앞서서 전송된다(즉, 포워드 된다). 그 결과, 마스터 회로(10)의 데이터 언로딩 동작은 데이터 로딩 동작 이후에 수행될 수 있게 된다. 그러므로, 슬레이브 회로(20)로부터 입력된 데이터(DATA_IN)는 마스터 회로(10)로 정확하게 전송될 수 있다.

클럭들의 포워딩을 위해서, 클럭 포워딩 회로(100)의 데이터 로딩/언로딩 초기 파라미터들(initial parameters)을 결정할 필요가 있다. 일반적으로, 이 초기 파라미터들은 마더보드 디자이너에 의해 미리 결정되고, 고정된 값으로 상기 클럭 포워딩 회로(100)에 연결된 외부 ROM(read only memory) (미 도시됨)에 저장된다. 전원이 인가된 후 마스터 회로(10)가 초기화 될 때, 초기 파라미터들은 클럭들의 포워딩을 위해 클럭 포워딩 회로(100)에 로딩된다. 이 초기 파라미터에 기초한 클럭 포워딩에 의해서, 슬레이브 회로(20)로부터 로딩된 입력 데이터는 아무런 오류도 발생하지 않고 마스터 회로(10)로 안전하게 언로딩 될 수 있다.

클럭 포워딩 방법에 대한 예는, 1989년 3월 7일 Sager et al.에 의해 등록된 U.S. Pat. No. 4,811,364, "METHOD AND APPARATUS FOR STABILIZED DATA TRANSMISSION", 1990년 12월 18일 Sager et al.에 의해 등록된 U.S. Pat. No. 4,979,190, "METHOD AND APPARATUS FOR STABILIZED DATA TRANSMISSION", 그리고 1985년 6월 25일 Wolf에 의해 등록된 U.S. Pat. No. 4,525,849, "DATA TRANSMISSION FACILITY BETWEEN TWO ASYNCHRONOUSLY CONTROLLED DATA PROCESSING SYSTEMS WITH A BUFFER MEMORY"에 있다.

앞에서 설명한 바와 같이, 클럭들을 포워딩 하기 위한 초기 파라미터들은, 마더보드 디자이너에 의해 제공되므로, 인건비의 증가에 비례해서 생산단가가 증가된다. 뿐만 아니라, 초기 파라미터들이 외부 ROM에 고정된 값으로 저장되기 때문에, 제품 편차에 의한 데이터 전송 오류가 발생할 수 있으므로, 회로들 사이의 데이터 전송을 안정화하기 어려운 문제가 있다.

발명이 이루고자 하는 기술적 과제

따라서, 본 발명의 목적은 상술한 제반 문제점을 해결하기 위해 제안된 것으로, 회로들 사이의 데이터 전송을 안정화하기 위해, 인터페이스 클럭들 사이의 딜레이를 자동으로 검출하고 데이터 로딩/언로딩 초기 파라미터들을 셋팅할 수 있는 디지털 정보처리 IC를 제공하는데 있다.

발명의 구성 및 작용

상술한 바와 같은 본 발명의 목적을 달성하기 위한 본 발명의 특징에 의하면, 디지털 시스템은, 클럭 포워딩 회로를 구비하고, 제 1 클럭 신호를 발생하기 위한 마스터 회로와, 상기 마스터 회로에 연결되어 상기 제 1 클럭 신호에 동기된 제 2 클럭 신호를 발생하기 위한 슬레이브 회로를 포함한다. 상기 클럭 포워딩 회로는 제 2 클럭 신호를 받아들이고, 제 1 및 제 2 클럭 신호들 사이의 딜레이를 검출하고, 그리고 상기 검출된 딜레이를 기초로 하여 상기 마스터 회로의 데이터 로드/언로드 초기 값을 셋팅한다.

(실시예)

이하 본 발명에 따른 실시예를 첨부된 도면 도 4 내지 도 11을 참조하여 상세히 설명한다.

본 발명의 신규한 클럭 포워딩 회로는 데이터 전송을 안정화하기 위해서, 인터페이스 클럭들이 가지고 있는 딜레이를 자동으로 검출하고, 검출된 딜레이에 상응하는 클럭들을 포워딩 한다(즉, 앞서 전송한다). 그 결과, 상기 회로들 사이의 데이터 전송은, 클럭들의 포워딩에 의해서 딜레이에 영향을 받지 않고 고품질 없이 수행될 수 있다.

도 4는 본 발명에 의한 마스터 회로의 클럭 포워딩 회로를 개략적으로 보여주기 위한 블록도이다. 도 4를 참조하면, 클럭 포워딩 회로(100)는 클럭 제너레이터(clock generator; 110), 출력 클럭 제어 로직(output clock control logic; 120), 내부 데이터 버스(internal data bus; 130), 입력 클럭 제어 로직(input clock control logic; 140), 로드/언로드 클럭 제어 로직(load/unload clock control logic; 150), 로드/언로드 멀티플렉서(load/unloading multiplexer; 160), 데이터 제어 로직(data control logic; 170), 그리고 딜레이 검출 회로(delay detection circuit; 180)를 포함한다.

클럭 제너레이터(110)는 데이터 입력 및 출력을 위한 비트 클럭을 발생한다. 출력 클럭 제어 로직(120)은, 클럭 제너레이터(110)로부터의 클럭 신호를 제어하여 출력 클럭 신호(CLK_OUT)를 슬레이브 회로로 발생한다. 이 때, 마스터 회로의 내부 데이터 버스(130)로부터 전송된 출력 데이터(DATA_OUT)는, 데이터 제어 로직(170)을 통해 슬레이브 회로로 출력된다.

상기 마스터 회로와 외부 인터페이스를 수행하는 슬레이브 회로는, 마스터 회로로부터의 클럭 신호(CLK_OUT)와 데이터(DATA_OUT)를 받아들이고, 입력 클럭 신호(CLK_IN)와 입력 데이터(DATA_IN)를 마스터 회로의 클럭 포워딩 회로(100)로 출력한다. 이 입력 클럭 신호(CLK_IN)는 출력 클럭 신호(CLK_OUT)의 피드백 클럭 신호(feedback clock signal)로서, 두 클럭 신호들(CLK_IN, CLK_OUT) 사이에는 딜레이가 존재한다. 이 딜레이는 마스터 회로와 슬레이브 회로를 포함하는 마더보드의 구조에 의해 발생된다.

상기 회로들 사이의 데이터 전송을 안정화하기 위해서, 딜레이 검출 회로(180)는 자동으로 상기 딜레이를 검출하고 딜레이에 대응되는 초기 파라미터(Init_UNLD)를 로드/언로드 클럭 제어 로직(150)으로 발생한다. 로드/언로드 클럭 제어 로직(150)은, 입력 클럭 신호(CLK_IN) 및 입력 클럭 제어 로직(140)으로부터 제어된 클럭 신호(CLK_OUT)에 응답해서, 다수 개의 로드 제어 신호들(LDs) 및 다수 개의 언로드 제어 신호들(UNLDs)을 로드/언로드 멀티플렉서(160)로 발생한다. 로드/언로드 멀티플렉서(160)에서 상기 로드 및 언로드 제어 신호들(LD, UNLD)은, 그것들의 초기 값에서부터 발생되기 시작된다. 언로드 제어 신호의 초기 값은 상기 초기 파라미터(Init_UNLD)에 의해서 셋팅 되고, 로드 제어 신호의 초기 값은 "00"으로 셋팅 된다. 이렇게 발생된 상기 로드 및 언로드 제어 신호들(LDs, UNLDs)에 의해서 로드/언로드 멀티플렉서(160)는 입력 데이터(DATA_IN)를 로딩할 수 있고, 그리고 로딩된 입력 데이터(DATA_IN)를 상기 딜레이에 영향을 받지 않고 데이터 제어 로직(170)을 통해 마스터 회로로 오류 없이 언로드할 수 있다.

도 5는 도 4에 도시된 딜레이 검출 회로를 보여주기 위한 회로도이다. 도 5를 참조하면, 딜레이 검출 회로(180)는 검출 유닛(188), 비교 유닛(195) 및 제어 유닛(198)을 포함한다.

검출 유닛(188)은 입력 클럭 신호(CLK_IN)와 출력 클럭 신호(CLK_OUT) 사이의 딜레이를 검출한다. 비교 유닛(195)은 N 개의 검출된 딜레이들을 비교하고, 모든 검출된 딜레이들이 일치하면 비교된 결과를 초기 파라미터(Init_UNLD)로서 로드/언로드 클럭 제어 로직(150)으로 출력한다. 반면, 만약 모든 검출된 딜레이들이 일치하지 않으면 제어 유닛(198)은 클럭 제너레이터(110) 및 검출 유닛(188)을 리셋 한다. 그리고 나서, 제어 유닛(198)은, 클럭 제너레이터(110)의 상태가 복원된 후(즉, 릴리즈(release)된 후) 모든 검출된 딜레이들이 일치할 때까지 N 비트 프리 런닝(N bit free running)에 의해 검출된 딜레이들을 비교한다.

상기 검출 유닛(188)은 카운팅 회로(181)와 딜레이 검출 회로(184)를 포함한다. 카운팅 회로(181)는 제 1 0-플립플롭(182)과 제 2 0-플립플롭(183)을 포함한다. 각각의 플립플롭은 마스터 회로로부터의 출력 클럭 신호(CLK_OUT)에 응답해서 토글링 동작을 수행한다. 그리고, 딜레이 검출 회로(184)는 제 3 0-플립플롭(185), 제 4 0-플립플롭(186) 및 R-S 플립플롭(187)을 포함한다. 제 1 0-플립플롭(182)의 출력은 제 3 0-플립플롭(185)의 입력과 연결되고, 제 2 0-플립플롭(183)의 출력은 제 4 0-플립플롭(186)의 입력과 연결된다. 제 3 0-플립플롭(185) 및 제 4 0-플립플롭(186)은 카운팅 회로(181)에서 카운팅 된 결과와 슬레이브 회로로부터 입력된 입력 클럭 신호(CLK_IN)에 응답해서 딜레이를 검출한다. 상기 입력 클럭 신호(CLK_IN)는 R-S 플립플롭(187)을 통해 제 3 0-플립플롭(185) 및 제 4 0-플립플롭(186)으로 클럭 신호로서 입력된다.

비교 유닛(195)은 래칭 회로(192)와 비교기(193)를 포함한다. 래칭 회로(192)는, N 비트 프리 런닝에 의해 검출 유닛(188)으로부터 검출된 딜레이를 각각의 MSB(most significant bit)와 LSB(least significant bit)를 래칭하기 위한 디멀티플렉서(de-multiplexer; 191)와 N개의 래치들을 포함한다. 비교기(193)는 두 개의 XOR(exclusive-OR) 게이트와 하나의 NOR 게이트를 가지는데, 각각의 XOR 게이트는 검출된 딜레이의 MSB N개와 LSB N개를 디멀티플렉서(191)를 통해 각각 받아들인다. XOR 게이트들은, 검출된 결과들이 모두 같은지 아닌지 검출된 결과들의 MSB들과 LSB들을 비교하고, 비교된 결과들을 NOR 게이트로 출력한다. 만약 검출된 결과들이 모두 같다면, NOR 게이트는 논리 하이 레벨("1")을 출력한다. 이 경우, 상기 MSB 및 LSB는 로드/언로드 클럭 제어 로직(150)에 초기 파라미터(Init_UNLD)로서 출력된다. 반면에, 만약 검출된 결과들이 모두 같지 않다면, NOR 게이트는 논리 로우 레벨("0")을 출력한다. 이 경우, 제어 유닛(198)은 올바른 딜레이가 검출되지 않았다고 판별하고 클럭 제너레이터(110) 및 검출 유닛(188)을 리

셋 한다.

제어 유닛(198)은 시스템 클럭 제어 로직(system clock control logic; 196)과 N 비트 프리 런닝 카운터/디코더(N bit free running counter/decoder; 197)를 포함한다. N 비트 프리 런닝 카운터/디코더(197)는 시스템 클럭 제어 로직(196)과 래칭 회로(192) 사이에 연결된다. N 비트 프리 런닝 카운터/디코더(197)는 상기 비교기(193)의 출력이 "0"일 때 인에이블 되어 N 비트 프리 런닝을 수행하고, 상기 클럭 제너레이터(110)로부터의 클럭 신호를 상기 래칭회로(192)의 래치클럭 공급하기 위해 디멀티플렉서(191)를 제어한다. 시스템 클럭 제어 로직(196)은 비교기(193), 검출 유닛(188), N 비트 프리 런닝 카운터/디코더(197), 그리고 클럭 제너레이터(110)에 연결된다. 만약 비교기(193)가 시스템 클럭 제어 로직(196)으로 "0"을 출력하면, 시스템 클럭 제어 로직(196)은 클럭 제너레이터(110)와 검출 유닛(188)을 리셋 한다. 리셋 된 클럭 제너레이터(110)가 정상 상태로 되돌아오면, 시스템 클럭 제어 로직(196)은 클럭 제너레이터(110)로부터의 클럭 신호를 N 비트 프리 런닝 카운터/디코더(197)로 전송한다.

도 6은 도 5에 도시된 딜레이 검출 회로의 동작을 보여주는 타이밍도이다. 도 5 및 도 6을 참조하면, 제 1 및 제 2 0 플립플롭들은 출력 클럭 신호(CLK_OUT)에 응답해서 토글링 동작을 수행한다. 파형 F/F <1>는 제 1 0 플립플롭(182)의 토글 결과이고, 파형 F/F <0>는 제 2 0 플립플롭(183)의 토글 결과이다. 딜레이 검출 회로(184)는, 토글 결과 F/F <1> 및 F/F <0>, 그리고 입력 클럭 신호(CLK_IN)에 응답해서 출력 클럭 신호(CLK_OUT)와 입력 클럭 신호(CLK_IN) 사이의 딜레이를 검출한다. 도 6에 도시된 바와 같이, 만약 최대 딜레이가 1 비트 타임(bit time) 보다 크고 2 비트 타임 보다 작다면, 검출된 딜레이는 "1"이 된다. 이는 입력 클럭 신호(CLK_IN)의 상승 에지(rising edge)에서의 토글 된 결과 F/F <0> 및 F/F <1>에 의해 결정된다. 그리고, 앞에서 설명한 바와 같이, N 개의 검출된 딜레이가 모두 같지 않으면, 시스템 클럭 제어 로직(196)은 리셋 신호(Reset)를 이용해 클럭 제너레이터(110)와 검출 유닛(188)을 리셋 한다. 리셋 된 클럭 제너레이터(110)가 정상 상태로 리셋 되면, 상기의 클럭 포워딩 방법은 검출된 딜레이들이 모두 같을 때까지 반복된다.

도 7은 도 5에 도시된 딜레이 검출 회로에 의해서 검출된 최대 딜레이에 대응되는 초기 파라미터를 보여주기 위한 타이밍도이다. 도 7을 참조하면, 초기 파라미터(Init_UNLD)는 검출된 최대 딜레이에 의해 결정된다. 도 6 및 도 7에 도시된 바와 같이, 만약 최대 딜레이가 1 비트 타임 보다 크고 2 비트 타임 보다 작다면, 초기 파라미터(Init_UNLD)는 "1"로 결정된다. 왜냐하면, 이 때 패치 된 결과 F <1>의 값이 "1"이기 때문이다. 그리고, 예를 들어, 만약 최대 딜레이가 1 보다 작다면, 초기 파라미터(Init_UNLD)는 "0"로 결정된다.

도 8은 도 4에 도시된 로드/언로드 클럭 제어 로직을 보여주는 타이밍도이다. 그리고 도 9는 도 8에 도시된 로드/언로드 클럭 제어 로직의 동작을 보여주는 타이밍도이다.

도 8을 참조하면, 로드/언로드 클럭 제어 로직(150)은 언로드 제어신호 발생회로(152) 및 로드 제어신호 발생회로(154)를 포함한다. 언로드 제어신호 발생회로(152)는, 클럭 제어 로직(140)에서 제어된 클럭 신호(CLK_OUT)와 딜레이 검출회로(180)로부터 검출된 초기 파라미터(Init_UNLD)를 받아들여서 다수 개의 언로드 제어 신호들(UNLDs)을 발생하기 위한 두 개의 0 플립플롭들을 포함한다. 언로드 제어 신호(UNLD)의 초기 값은 초기 파라미터(Init_UNLD)에 의해 셋팅된다. 이와 유사하게, 로드 제어신호 발생회로(154)는, 슬레이브 회로로부터의 입력 클럭 신호(CLK_IN)에 응답해서 다수 개의 로드 제어 신호들(LDs)을 발생하기 위한 두 개의 0 플립플롭들을 포함하며, 로드 제어 신호(LD)의 초기 값은 "00"으로 셋팅된다.

도 9를 참조하면, 만약 언로드 제어 신호(UNLD)의 초기 값이 "11"로 셋팅되면, 언로드 제어 신호(UNLD)는 상기 클럭 신호(CLK_OUT)에 응답해서 "10", "00", "01", 그리고 "11"의 순서로 반복해서 발생된다. 그러나, 로드 제어 신호(LD)는 항상 입력 클럭 신호(CLK_IN)에 응답해서 "01", "11", "10", 그리고 "00"의 순서로 반복해서 발생된다.

도 10은 도 4에 도시된 로드/언로드 멀티플렉서를 보여주는 타이밍도이다. 도 10을 참조하면, 로드/언로드 멀티플렉서(160)는 데이터 로딩 회로(161)와 데이터 언로딩 회로(166)를 포함한다. 데이터 로딩 회로(161)는 데이터 로딩 디코더(162)와 다수 개의 데이터 로딩 멀티플렉서들(163)을 포함한다. 데이터 언로딩 회로(166)는 데이터 언로딩 디코더(167)와 다수 개의 데이터 언로딩 멀티플렉서들(168)을 포함한다. 이 경우에 있어서, 입력되는 로드 제어 신호(LD) 및 언로드 제어 신호(UNLD)는 각각 4개씩이므로, 데이터 로딩 멀티플렉서들(163) 및 데이터 언로딩 멀티플렉서들(168)은 각각 4개의 멀티플렉서들로 구성된다.

각각의 데이터 로딩 멀티플렉서들(163)은, 디코딩 된 로드 제어 신호(LD)에 응답해서 슬레이브 회로로부터의 입력 데이터(DATA_IN)를 로딩한다. 마찬가지로, 각각의 데이터 언로딩 멀티플렉서들(168)은, 디코딩 된 언로드 제어 신호(UNLD)에 응답해서 로딩된 입력 데이터(DATA_IN)를 마스터 회로의 내부 데이터 버스로 언로딩 한다.

도 11은 도 4에 도시된 클럭 포워딩 회로의 동작을 보여주는 타이밍도이다. 도 10 및 도 11을 참조하면, 만약 언로드 컨트롤 신호(UNLD)의 초기 값이 "11"로 셋팅 되었다면, 언로드 컨트롤 신호(UNLD)는 출력 클럭 신호(CLK_OUT)에 응답해서 "10", "00", "01", 그리고 "11" 순서로 반복적으로 발생된다. 그리고 로드 컨트롤 신호(LD)는 입력 클럭 신호(CLK_IN)에 응답해서 "01", "11", "10", 그리고 "00" 순서로 반복적으로 발생된다. 이 때, 멀티플렉서들(163, 168)에 속한 각각의 멀티플렉서들은, 입력 데이터(DATA_IN)를 로드/언로드 하기 위해 로드 컨트롤 신호(LD) 및 언로드 컨트롤 신호(UNLD)에 응답해서 턴 온 된다.

만약 언로드 컨트롤 신호(UNLD)의 초기 값이 "11"로 셋팅 되었다면, 이는 최대 딜레이가 1 비트 타임 보다 크고 2 비트 타임보다 작다는 것을 의미한다. 따라서, 본 발명에 의한 클럭 포워딩 회로는, 두 회로 사이의 데이터 전송을 딜레이에 영향을 받지 않고 안정화시키기 위해 언로드 제어 신호들(UNLDs) "10" 및 "00"에 대응되는 두 클럭을 포워딩 한다(즉, 미리 전송한다).

그러므로, 클럭들의 포워딩 후 각각의 데이터 로딩 멀티플렉서(163)는, "01", "11", "10" 및 "00" 순서의 로드 제어 신호(LD)에 응답해서 입력 데이터(DATA_IN)를 로딩하면, 데이터 언로딩 멀티플렉서(168)는, "01", "11", "10" 및 "00" 순서의 언로드 제어 신호(UNLD)에 응답해서 로딩된 입력 데이터(DATA_IN)를 로

당한다. 예를 들면, 로드 제어 신호(LD) "이"에 의해서 로딩된 입력 데이터(DATA_IN)는, 언로딩 제어 신호(UNLD) "이"에 의해서 마스터 회로로 언로딩 되어 딜레이에 영향을 받지 않고 데이터를 로딩/언로딩 할 수 있게 된다.

그 결과, 본 발명에 의한 클럭 포워딩 회로는, 인터페이스 클럭들 사이의 딜레이를 자동으로 검출하고 그 검출된 딜레이에 상응하는 클럭들을 포워딩 함으로써, 인터페이스 클럭들의 딜레이에 영향을 받지 않고 회로를 시아의 데이터 전송을 오류 없이 수행할 수 있다.

이상에서, 본 발명에 따른 회로의 구성 및 동작을 상기한 설명 및 도면에 따라 도시하였지만 이는 예를 들어 설명한 것에 불과하며 본 발명의 기술적 사상을 벗어나지 않는 범위 내에서 다양한 변화 및 변경이 가능함은 물론이다.

발명의 효과

이와 같은 장치에 의해서, 회로들 사이의 인터페이스 클럭들 사이의 딜레이를 자동으로 검출하고 그 검출된 딜레이에 상응하는 클럭들을 포워딩 함으로써, 인터페이스 클럭들의 딜레이에 영향을 받지 않고 회로들 사이의 데이터 전송을 오류 없이 수행할 수 있다.

(5) 청구의 범위

청구항 1

디지털 시스템에 있어서:

클럭 포워딩 회로를 포함하고, 제 1 클럭 신호를 발생하기 위한 마스터 회로; 그리고

상기 마스터 회로에 연결되어, 상기 제 1 클럭 신호에 동기된 제 2 클럭 신호를 발생하기 위한 슬레이브 회로를 포함하며,

상기 클럭 포워딩 회로는 제 2 클럭 신호를 받아들이고, 제 1 및 제 2 클럭 신호들 사이의 딜레이를 검출하고, 그리고 상기 검출된 딜레이를 기초로 하여 상기 마스터 회로의 데이터 로드/언로드 초기 값을 설정하는 것을 특징으로 하는 자동 클럭 딜레이 검출 및 초기 파라미터 셋팅 특성을 가진 클럭 포워딩 회로.

청구항 2

클럭 포워딩 회로에 있어서:

클럭 신호를 발생하기 위한 클럭 제너레이터와;

마스터 회로의 데이터 인터페이스를 위한 내부 데이터 버스와;

상기 클럭 제너레이터와 상기 내부 데이터 버스에 연결되고, 상기 클럭 신호에 응답해서 데이터를 슬레이브 회로로 출력하기 위한 데이터 제어 로직과;

상기 클럭 제너레이터에 연결되어, 상기 클럭 신호를 제어하여 제어된 클럭 신호를 출력 클럭 신호로서 상기 슬레이브 회로로 발생하기 위한 출력 클럭 제어 로직과;

상기 출력 클럭 신호와 상기 출력 클럭 신호의 피드백 클럭 신호인 입력 클럭 신호 사이의 딜레이를 검출하고, 그리고 검출된 딜레이에 상응하는 클럭들을 포워딩하기 위한 초기 파라미터를 발생하기 위한 딜레이 검출 회로와;

상기 딜레이 검출 회로와 상기 입력 클럭 제어 로직에 연결되고, 상기 초기 파라미터에 응답해서 로드 제어 신호들과 언로드 제어 신호들을 발생하기 위한 로드/언로드 클럭 제어 로직과;

상기 클럭 제너레이터와 상기 로드/언로드 클럭 제어 로직 사이에 연결되어, 상기 클럭 제너레이터로부터의 상기 클럭 신호를 제어하여 제어된 클럭 신호를 상기 로드/언로드 클럭 제어 로직에 공급하기 위한 입력 클럭 제어 로직; 그리고

상기 슬레이브 회로로부터의 입력 데이터를 로딩하고, 그리고 상기 로드/언로드 클럭 제어 로직으로부터의 로드 및 언로드 제어 신호들에 응답해서 상기 로딩된 입력 데이터를 상기 데이터 제어 로직을 통해 상기 마스터 회로의 상기 내부 데이터 버스로 언로딩하기 위한 로드/언로드 멀티플렉서를 포함하는 것을 특징으로 하는 자동 클럭 딜레이 검출 및 초기 파라미터 셋팅 특성을 가진 클럭 포워딩 회로.

청구항 3

제 2 항에 있어서,

상기 딜레이 검출 회로는,

상기 출력 클럭 신호와 상기 입력 클럭 신호 사이의 상기 딜레이를 검출하기 위한 검출 수단과;

상기 검출된 딜레이들을 비교하고, 모든 검출된 딜레이들이 일치할 때 비교된 결과를 상기 초기 파라미터로서 상기 로드/언로드 클럭 제어 로직으로 발생하기 위한 비교 수단; 그리고

상기 검출된 딜레이들이 모두 같지 않으면 상기 클럭제너레이터와 상기 검출 수단을 리셋하고, 그리고 상기 검출된 딜레이들이 모두 같을 때까지 N 비트 프리 런닝에 의해 상기 검출된 딜레이들을 비교하도록 상기 비교 수단을 제어하기 위한 제어 수단을 포함하며,

상기 검출 수단은 N 개의 딜레이를 검출하는 것을 특징으로 하는 자동 클럭 딜레이 검출 및 초기 파라미터 셋팅 특성을 가진 클럭 포워딩 회로.

청구항 4

제 3 항에 있어서,

상기 검출 수단은,

두 개의 D 플립플롭으로 이루어진 카운팅 유닛; 그리고

상기 카운팅 유닛으로부터 카운트 된 결과와 슬레이브 회로부터의 입력 클럭 신호에 응답해서 상기 딜레이를 검출하기 위한 검출 유닛을 포함하며,

상기 카운팅 유닛의 각각의 플립플롭은 상기 마스터 회로부터의 클럭 클럭 신호에 응답해서 토글링하는 것을 특징으로 하는 자동 클럭 딜레이 검출 및 초기 파라미터 셋팅 특성을 가진 클럭 포워딩 회로.

청구항 5

제 3 항에 있어서,

상기 비교 수단은,

디멀티플렉서와 N 개의 래치를 구비하여, N 비트 프리 런닝에 의해 상기 검출 수단으로부터 검출된 딜레이들의 MSB(most significant bit) 및 LSB(least significant bit) 각각을 래칭하기 위한 래칭 유닛; 그리고

상기 래칭 유닛으로부터의 MSB 및 LSB를 각각을 비교하고, 그리고 모든 MSB들과 모든 LSB들이 각기 같을 때 상기 MSB들 중 어느 하나와 LSB들 중 어느 하나를 상기 초기 파라미터로서 상기 로드/언로드 제어 로직으로 출력하기 위한 비교 유닛을 포함하는 것을 특징으로 하는 자동 클럭 딜레이 검출 및 초기 파라미터 셋팅 특성을 가진 클럭 포워딩 회로.

청구항 6

제 5 항에 있어서,

상기 제어 수단은,

N 비트 프리 런닝을 수행하기 위해 상기 디멀티플렉서를 제어하고, 그리고 상기 클럭 제너레이터로부터의 클럭 신호를 상기 래칭 유닛으로 공급하기 위한 N 비트 프리 런닝 카운터/디코더; 그리고

상기 클럭 제너레이터와 상기 검출 수단을 리셋하고, 그리고 검출된 결과들이 모두 같지 않을 때 상기 N 비트 프리 런닝 카운터/디코더를 인에이블시키기 위한 시스템 클럭 제어 로직을 포함하며,

상기 시스템 클럭 제어 로직은 상기 클럭 제너레이터가 정상 상태로 릴리즈 된 후 상기 클럭 신호를 상기 N 비트 프리 런닝 카운터/디코더로 공급하는 것을 특징으로 하는 자동 클럭 딜레이 검출 및 초기 파라미터 셋팅 특성을 가진 클럭 포워딩 회로.

청구항 7

제 2 항에 있어서,

상기 로드/언로드 제어 로직은,

상기 슬레이브 회로부터의 상기 입력 클럭 신호에 응답해서 다수 개의 로드 제어 신호들을 발생하기 위한 로드 제어 신호 발생 수단; 그리고

상기 클럭 제어 로직으로부터의 제어된 클럭 신호에 응답해서 다수 개의 언로드 제어 신호들을 발생하기 위한 언로드 제어 신호 발생 수단을 포함하는 것을 특징으로 하는 자동 클럭 딜레이 검출 및 초기 파라미터 셋팅 특성을 가진 클럭 포워딩 회로.

청구항 8

제 2 항에 있어서,

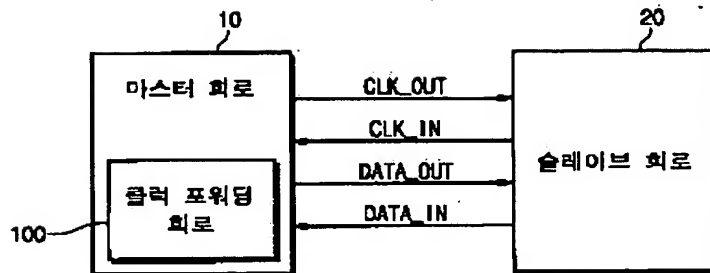
상기 로드/언로드 멀티플렉서는,

데이터 로딩 디코더와 다수 개의 멀티플렉서들을 포함하고, 상기 로드 제어 신호들에 응답해서 상기 슬레이브 회로부터 전송된 상기 입력 데이터를 로딩하기 위한 데이터 로딩 회로; 그리고

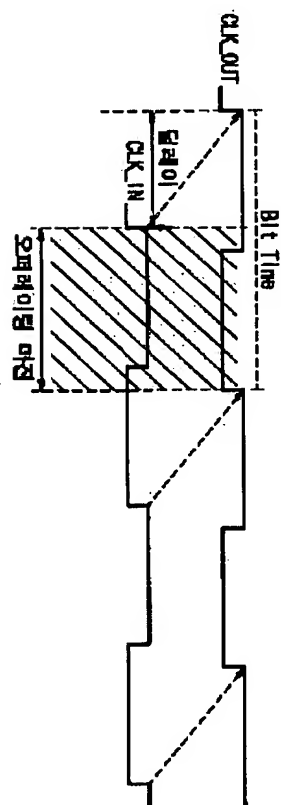
데이터 언로딩 디코더와 다수 개의 멀티플렉서들을 포함하고, 상기 언로드 제어 신호들에 응답해서 상기 로딩된 입력 데이터를 상기 데이터 제어 로직을 통해 상기 마스터 회로로 언로딩하기 위한 데이터 언로딩 회로를 포함하는 것을 특징으로 하는 자동 클럭 딜레이 검출 및 초기 파라미터 셋팅 특성을 가진 클럭 포워딩 회로.

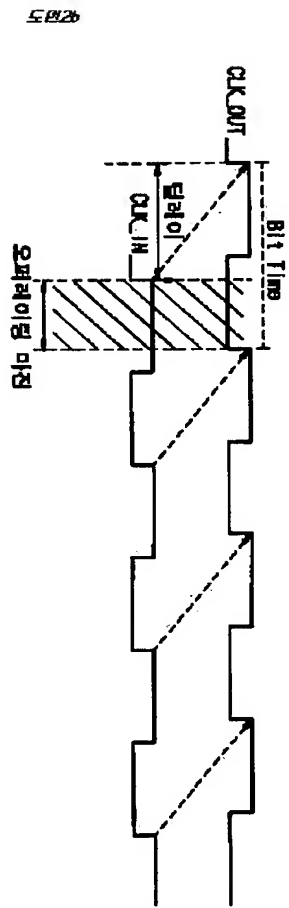
도면

도면1

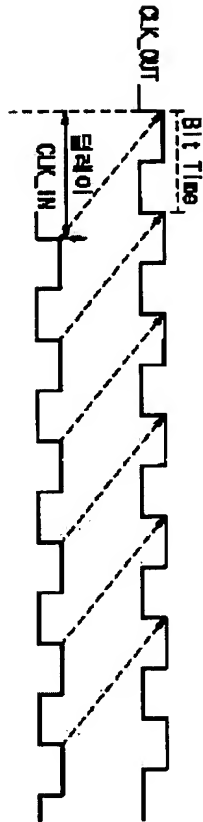


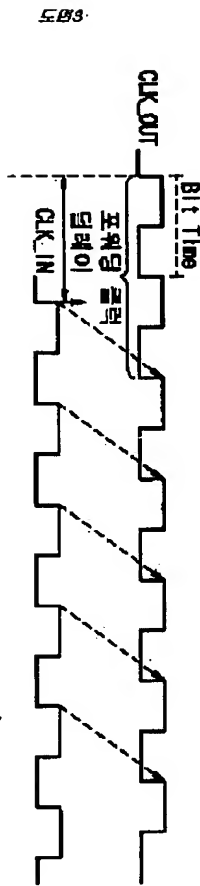
도면2a



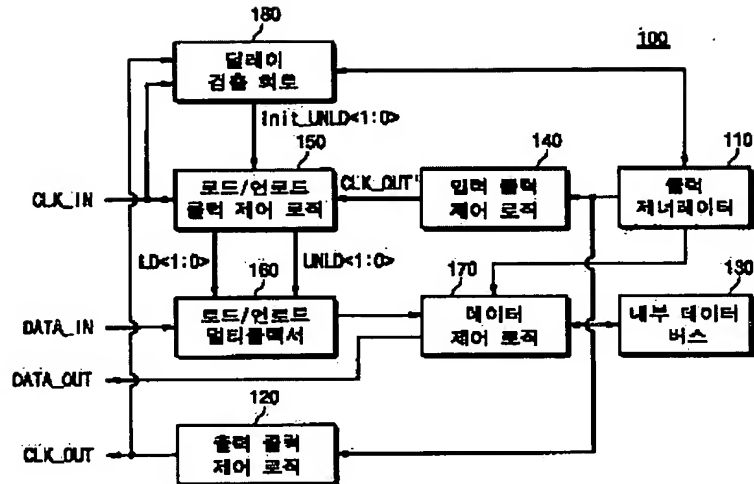


도 B2b

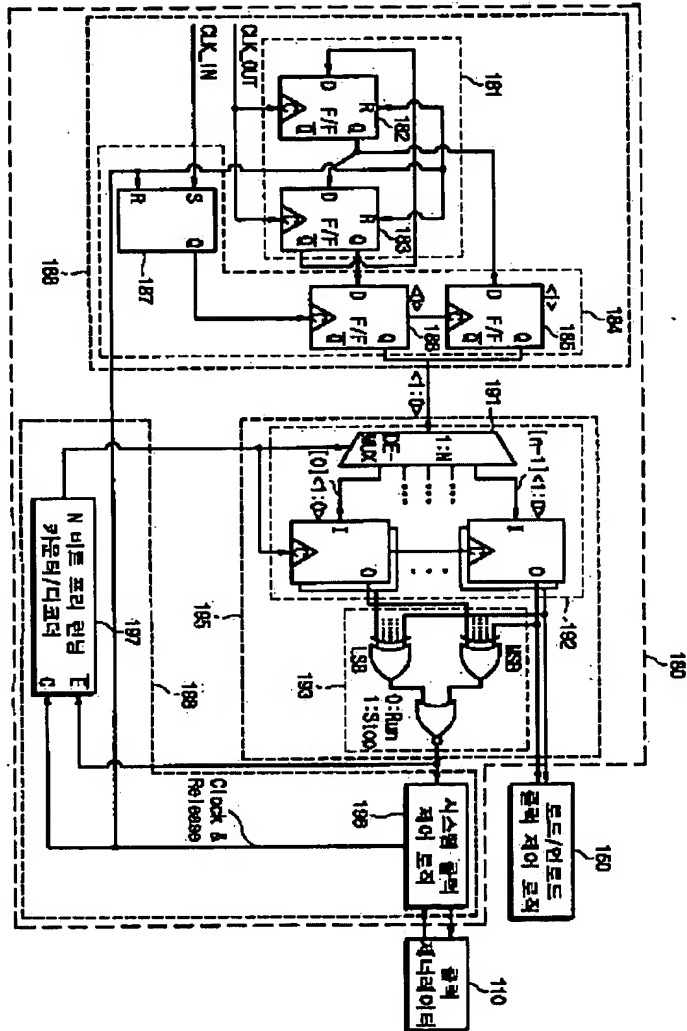




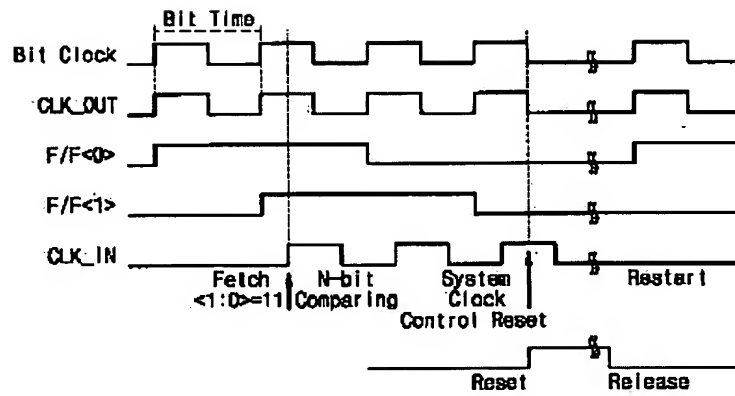
도 14



도 185



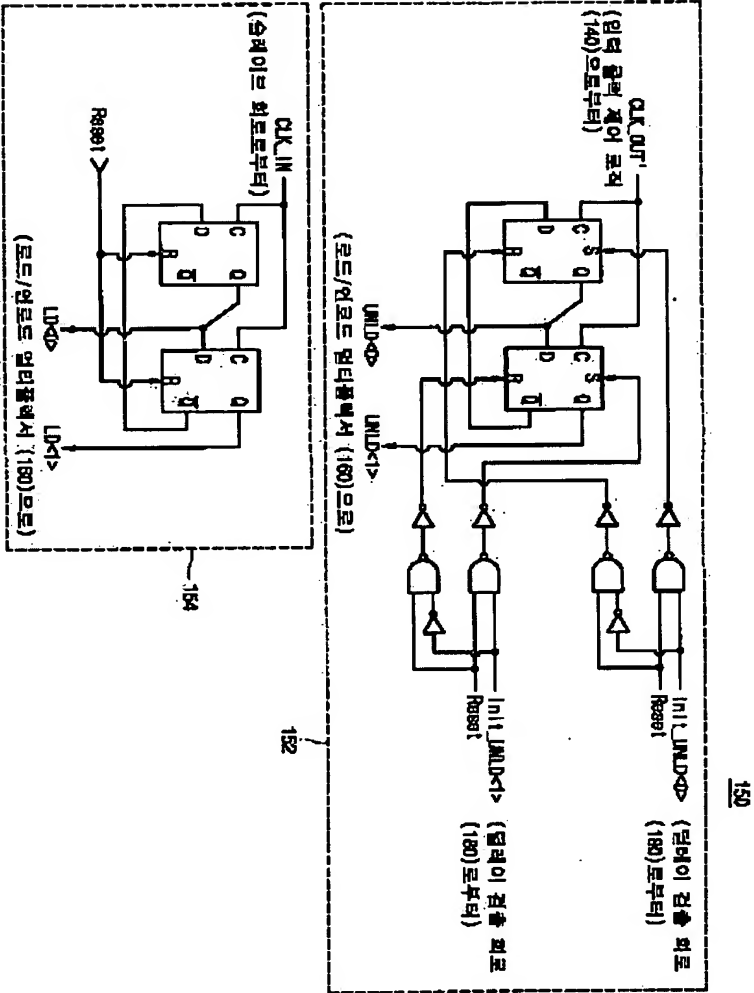
도 88



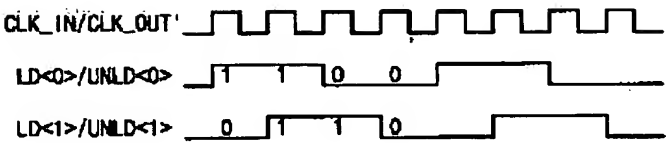
도 89

최대 딜레이	Init_UNLD<1:0>
$MD < 1 \text{ Bit Time}$	01
$1 < MD \leq 2 \text{ Bit Time}$	11
$2 < MD \leq 3 \text{ Bit Time}$	10
$3 < MD \leq 4 \text{ Bit Time}$	00

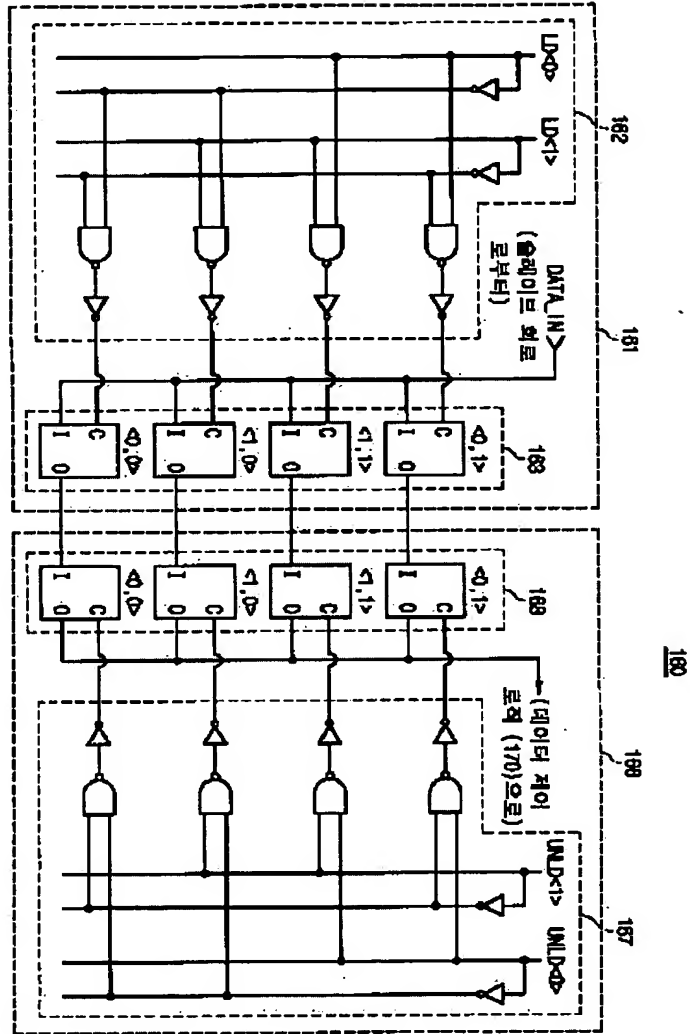
도면 8



도 19



도면 10



도면 11

